NATIONAL INSTITUTE OF TECHNOLOGY KARNATAKA
SURATHKAL, MANGALORE - 575 025

Course Code – CS254
Course Name – Database Systems Lab

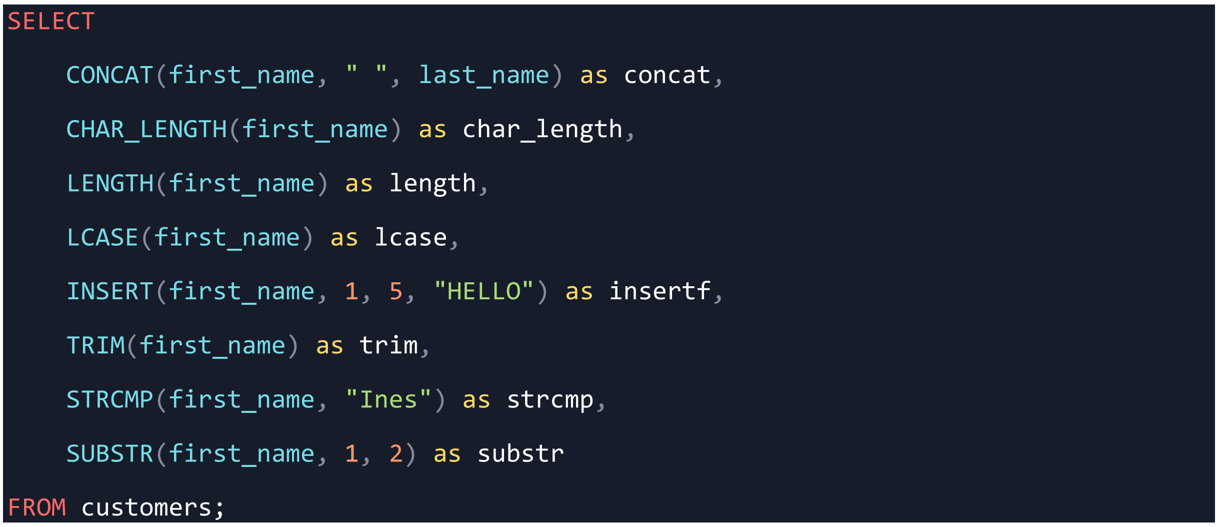
Lab - 05
Date – March 2, 2022

Submitted To
Dr. Annappa B
Department of Computer Science and Engineering
National Institute of Technology Karnataka, Surathkal

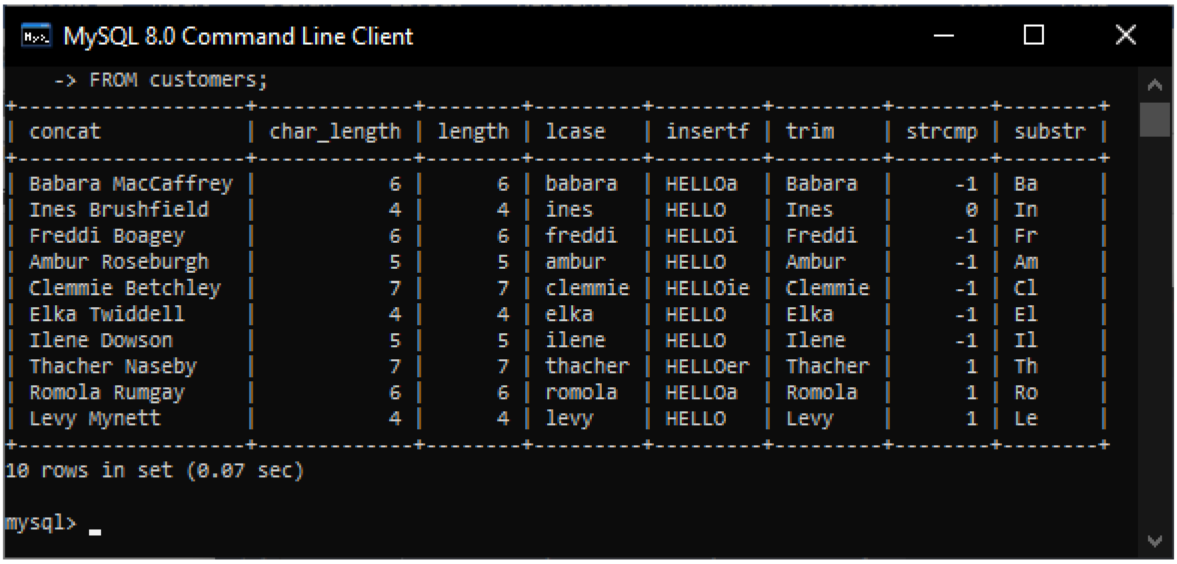
Submitted By
Md Rakib Hasan
Roll – 201CS132
Department of Computer Science and Engineering

1. Build a basic database (of your choice) and explore the usage of following string function:

CHAR_LENGTH()
CONCAT()
INSERT()
LCASE()
LENGTH()
LIKE
TRIM()
STRCMP()
SUBSTR()



```
SELECT
    CONCAT(first_name, " ", last_name) as concat,
    CHAR_LENGTH(first_name) as char_length,
    LENGTH(first_name) as length,
    LCASE(first_name) as lcase,
    INSERT(first_name, 1, 5, "HELLO") as insertf,
    TRIM(first_name) as trim,
    STRCMP(first_name, "Ines") as strcmp,
    SUBSTR(first_name, 1, 2) as substr
FROM customers;
```

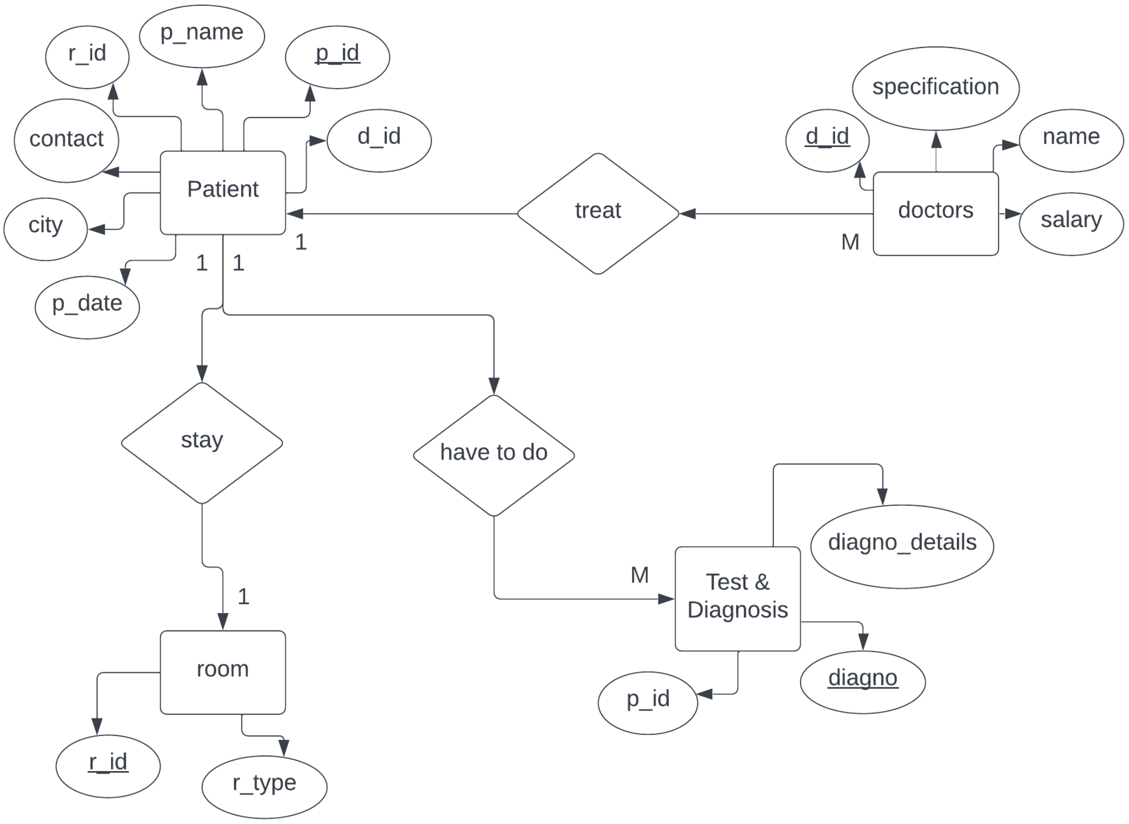


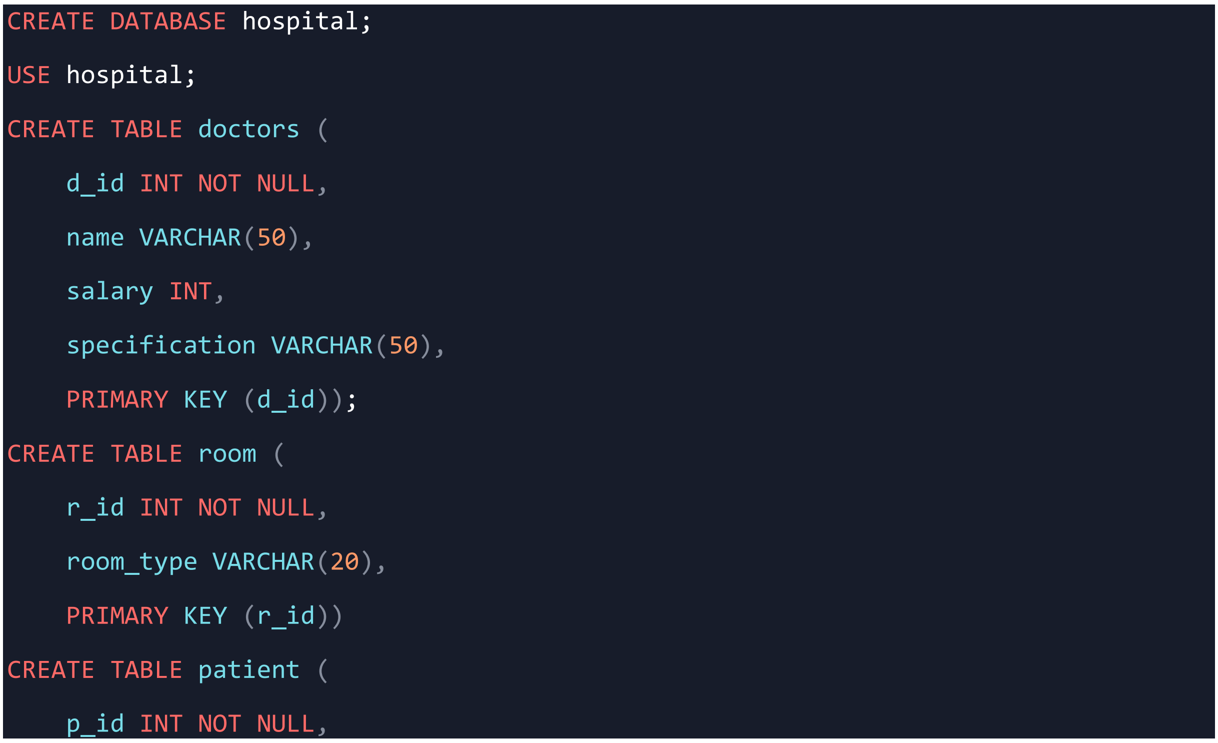
```
    -> FROM customers;
+-------------------+-------------+--------+---------+---------+---------+--------+--------+
| concat            | char_length | length | lcase   | insertf | trim    | strcmp | substr |
+-------------------+-------------+--------+---------+---------+---------+--------+--------+
| Babara MacCaffrey |           6 |      6 | babara  | HELLOa  | Babara  |     -1 | Ba     |
| Ines Brushfield   |           4 |      4 | ines    | HELLO   | Ines    |      0 | In     |
| Freddi Boagey     |           6 |      6 | freddi  | HELLOi  | Freddi  |     -1 | Fr     |
| Ambur Roseburgh   |           5 |      5 | ambur   | HELLO   | Ambur   |     -1 | Am     |
| Clemmie Betchley  |           7 |      7 | clemmie | HELLOie | Clemmie |     -1 | Cl     |
| Elka Twiddell     |           4 |      4 | elka    | HELLO   | Elka    |     -1 | El     |
| Ilene Dowson      |           5 |      5 | ilene   | HELLO   | Ilene   |     -1 | Il     |
| Thacher Naseby    |           7 |      7 | thacher | HELLOer | Thacher |      1 | Th     |
| Romola Rumgay     |           6 |      6 | romola  | HELLOa  | Romola  |      1 | Ro     |
| Levy Mynett       |           4 |      4 | levy    | HELLO   | Levy    |      1 | Le     |
+-------------------+-------------+--------+---------+---------+---------+--------+--------+
10 rows in set (0.07 sec)

mysql> _
```

2. Create database with

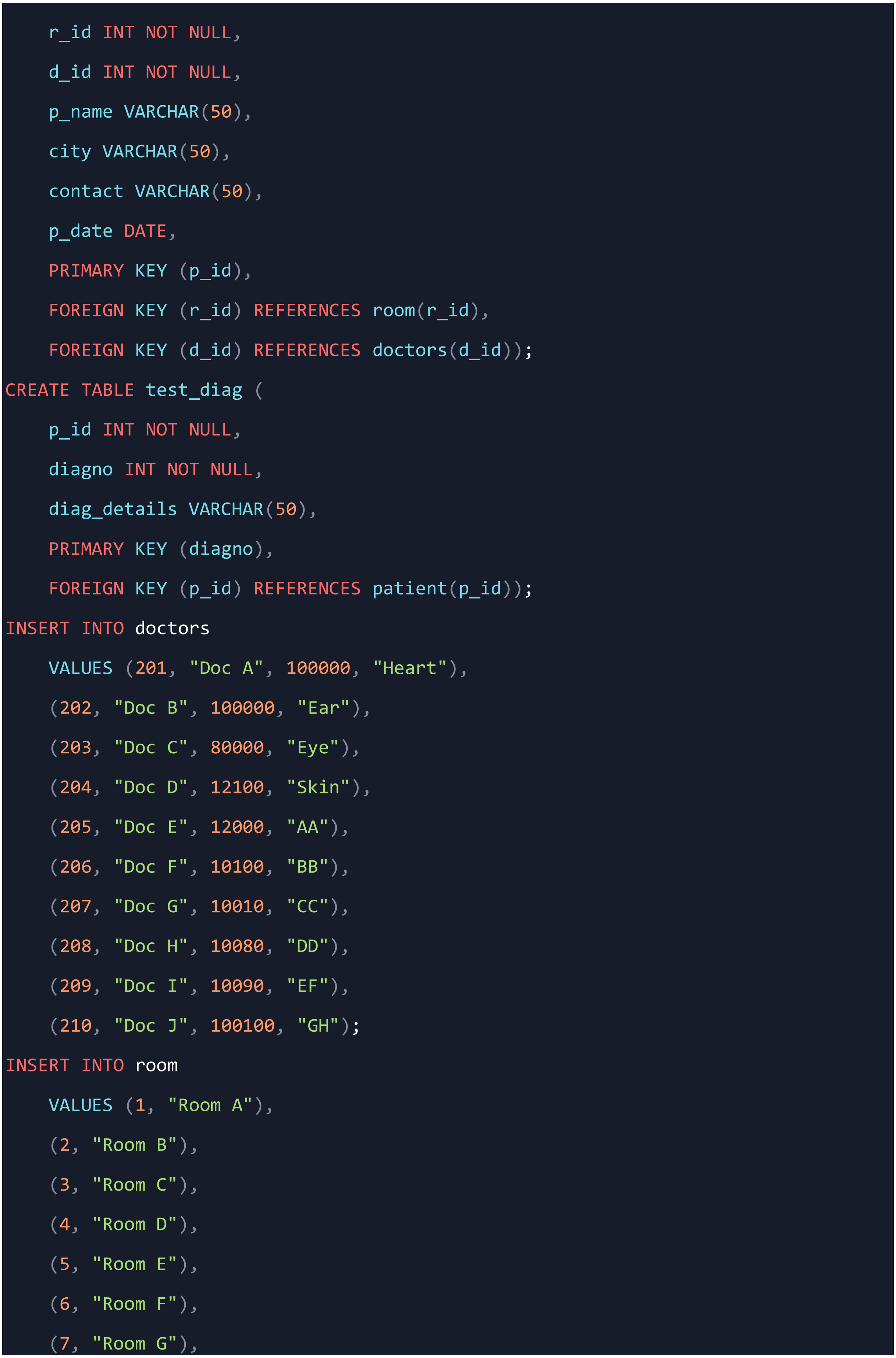
PATIENT (p_id, r_id, d_id, p_name, city, contact, p_date),
DOCTORS (d_id, name, salary, specification),
ROOM (r_id, room_type)
TEST & DIAGNOSIS (p_id, diagno, diag_details),

(Insert new five values for each table. Assume the necessary values related to below mentioned questions.)
(Add 10 entries for each table)
Draw the ER diagram for the above database.





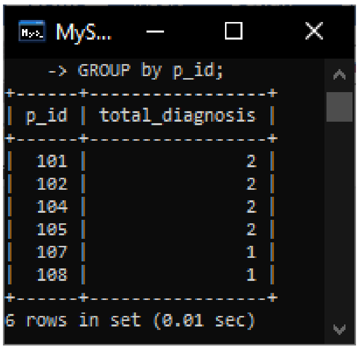
```
CREATE DATABASE hospital;
USE hospital;
CREATE TABLE doctors (
    d_id INT NOT NULL,
    name VARCHAR(50),
    salary INT,
    specification VARCHAR(50),
    PRIMARY KEY (d_id));
CREATE TABLE room (
    r_id INT NOT NULL,
    room_type VARCHAR(20),
    PRIMARY KEY (r_id))
CREATE TABLE patient (
    p_id INT NOT NULL,
```



```
    r_id INT NOT NULL,
    d_id INT NOT NULL,
    p_name VARCHAR(50),
    city VARCHAR(50),
    contact VARCHAR(50),
    p_date DATE,
    PRIMARY KEY (p_id),
    FOREIGN KEY (r_id) REFERENCES room(r_id),
    FOREIGN KEY (d_id) REFERENCES doctors(d_id));
CREATE TABLE test_diag (
    p_id INT NOT NULL,
    diagno INT NOT NULL,
    diag_details VARCHAR(50),
    PRIMARY KEY (diagno),
    FOREIGN KEY (p_id) REFERENCES patient(p_id));
INSERT INTO doctors
    VALUES (201, "Doc A", 100000, "Heart"),
    (202, "Doc B", 100000, "Ear"),
    (203, "Doc C", 80000, "Eye"),
    (204, "Doc D", 12100, "Skin"),
    (205, "Doc E", 12000, "AA"),
    (206, "Doc F", 10100, "BB"),
    (207, "Doc G", 10010, "CC"),
    (208, "Doc H", 10080, "DD"),
    (209, "Doc I", 10090, "EF"),
    (210, "Doc J", 100100, "GH");
INSERT INTO room
    VALUES (1, "Room A"),
    (2, "Room B"),
    (3, "Room C"),
    (4, "Room D"),
    (5, "Room E"),
    (6, "Room F"),
    (7, "Room G"),
    (8, "Room H"),
    (9, "Room I"),
    (10, "Room J");
INSERT INTO patient
    VALUES (101, 1, 201, "Patient A", "Dhaka", "01521", "2010-01-01"),
    (102, 2, 202, "Patient B", "Tangail", "01521001", "2012-01-01"),
    (103, 3, 201, "Patient C", "Rajshahi", "0152147", "2013-01-01"),
    (104, 4, 203, "Patient D", "Mymensingh", "01520111", "2014-01-01"),
    (105, 5, 203, "Patient E", "Chandpur", "0152146", "2017-01-01"),
    (106, 6, 201, "Patient F", "Cumilla", "01146461", "2010-02-01"),
    (107, 7, 204, "Patient G", "Kolkata", "046421", "2010-05-01"),
    (108, 8, 201, "Patient H", "Dilhi", "01589", "2010-09-01"),
    (109, 9, 205, "Patient I", "Karnataka", "01571", "2010-04-01"),
    (110, 10, 206, "PatientJA", "Mangalore", "0821", "2010-07-01");
INSERT INTO test_diag
    VALUES (101, 501, "ECG"), (102, 502, "XA"), (102, 503, "XB"),
    (104, 504, "XC"),
    (105, 505, "XD"),
    (105, 506, "XE"),
    (107, 507, "XF"),
    (108, 508, "XI"),
    (104, 509, "XJ"),
    (101, 510, "XK");
```

List the patient details with multiple diagnosis records.

```
SELECT p_id, COUNT(*) AS total_diagnosis
FROM test_diag
GROUP by p_id
```

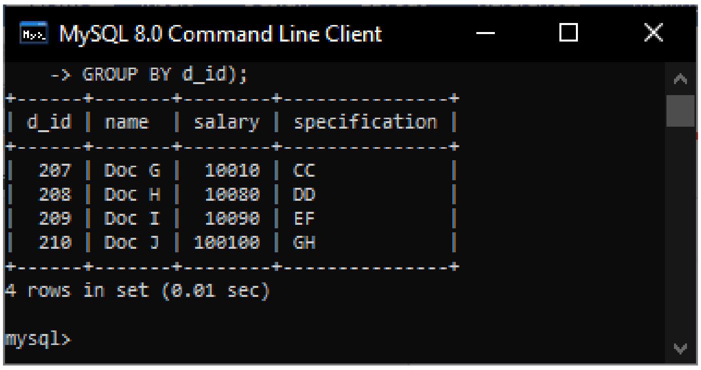


```
    -> GROUP by p_id;
+------+-----------------+
| p_id | total_diagnosis |
+------+-----------------+
|  101 |               2 |
|  102 |               2 |
|  104 |               2 |
|  105 |               2 |
|  107 |               1 |
|  108 |               1 |
+------+-----------------+
6 rows in set (0.01 sec)
```

Add a new attribute p_date (i.e hospital joining date) to the PATIENT table.

Fetch the doctors who do not have any patients.

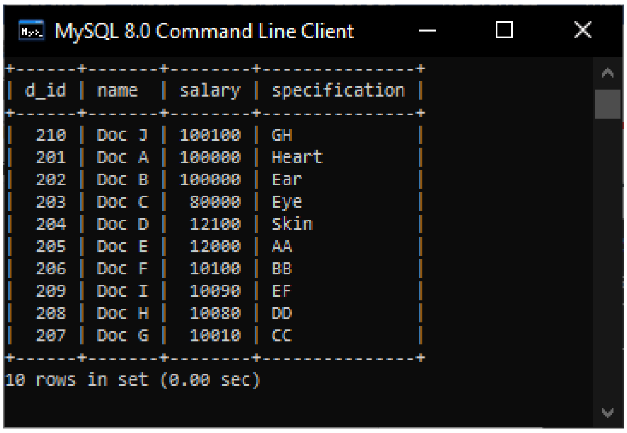
```
SELECT *
FROM doctors
WHERE d_id NOT IN (
    SELECT d_id
    FROM patient
    GROUP BY d_id)
```



```
    -> GROUP BY d_id);
+------+-------+--------+---------------+
| d_id | name  | salary | specification |
+------+-------+--------+---------------+
|  207 | Doc G |  10010 | CC            |
|  208 | Doc H |  10080 | DD            |
|  209 | Doc I |  10090 | EF            |
|  210 | Doc J | 100100 | GH            |
+------+-------+--------+---------------+
4 rows in set (0.01 sec)

mysql>
```

Display doctors' salary in ascending order.

```
SELECT *
FROM doctors
ORDER BY salary DESC
```

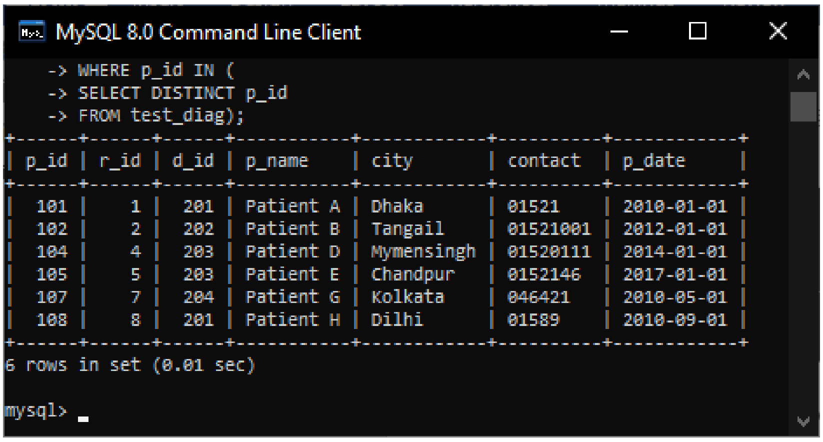


```
+------+-------+--------+---------------+
| d_id | name  | salary | specification |
+------+-------+--------+---------------+
|  210 | Doc J | 100100 | GH            |
|  201 | Doc A | 100000 | Heart         |
|  202 | Doc B | 100000 | Ear           |
|  203 | Doc C |  80000 | Eye           |
|  204 | Doc D |  12100 | Skin          |
|  205 | Doc E |  12000 | AA            |
|  206 | Doc F |  10100 | BB            |
|  209 | Doc I |  10090 | EF            |
|  208 | Doc H |  10080 | DD            |
|  207 | Doc G |  10010 | CC            |
+------+-------+--------+---------------+
10 rows in set (0.00 sec)
```

Display each patient details through diagd_details.

```
SELECT *
FROM patient
```

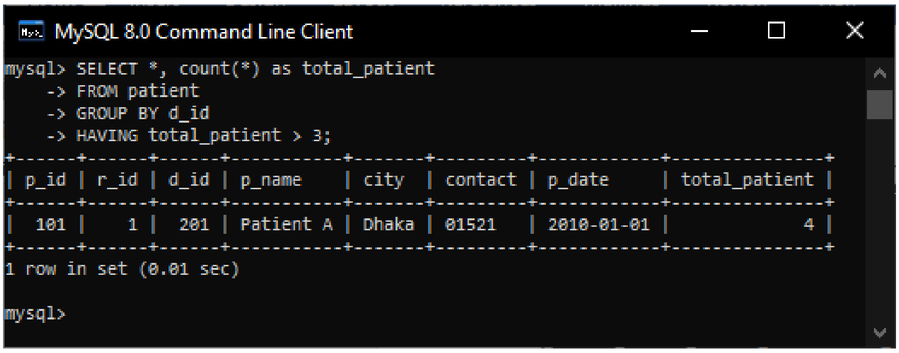
```
WHERE p_id IN (
    SELECT DISTINCT p_id
    FROM test_diag)
```



```
    -> WHERE p_id IN (
    -> SELECT DISTINCT p_id
    -> FROM test_diag);
+------+------+------+-----------+------------+----------+------------+
| p_id | r_id | d_id | p_name    | city       | contact  | p_date     |
+------+------+------+-----------+------------+----------+------------+
|  101 |    1 |  201 | Patient A | Dhaka      | 01521    | 2010-01-01 |
|  102 |    2 |  202 | Patient B | Tangail    | 01521001 | 2012-01-01 |
|  104 |    4 |  203 | Patient D | Mymensingh | 01520111 | 2014-01-01 |
|  105 |    5 |  203 | Patient E | Chandpur   | 0152146  | 2017-01-01 |
|  107 |    7 |  204 | Patient G | Kolkata    | 046421   | 2010-05-01 |
|  108 |    8 |  201 | Patient H | Dilhi      | 01589    | 2010-09-01 |
+------+------+------+-----------+------------+----------+------------+
6 rows in set (0.01 sec)

mysql> _
```

Display the number of patients for each doctor. Only include doctors with more than 3 patients.

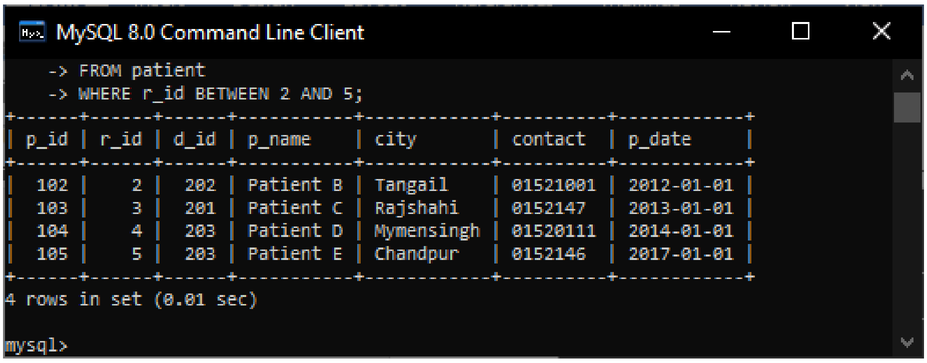
```
SELECT *, count(*) as total_patient
FROM patient
GROUP BY d_id
HAVING total_patient > 3
```



```
mysql> SELECT *, count(*) as total_patient
    -> FROM patient
    -> GROUP BY d_id
    -> HAVING total_patient > 3;
+------+------+------+-----------+-------+---------+------------+---------------+
| p_id | r_id | d_id | p_name    | city  | contact | p_date     | total_patient |
+------+------+------+-----------+-------+---------+------------+---------------+
|  101 |    1 |  201 | Patient A | Dhaka | 01521   | 2010-01-01 |             4 |
+------+------+------+-----------+-------+---------+------------+---------------+
1 row in set (0.01 sec)

mysql>
```

Display the doctors who are treating patients from r_id 102 to 105.

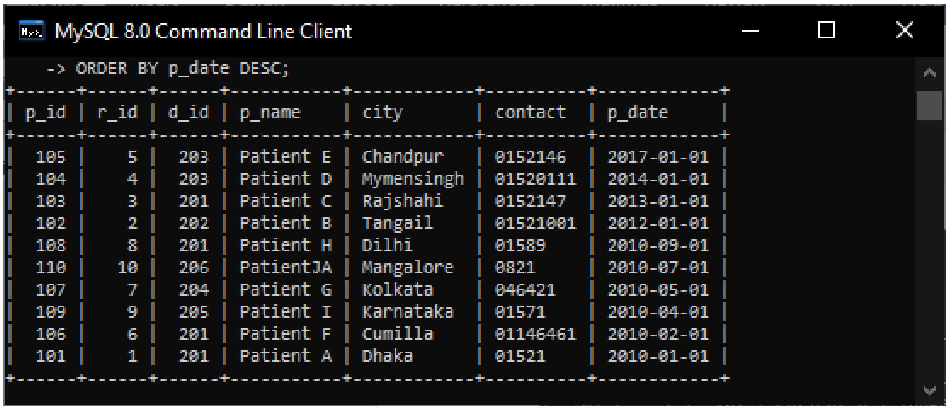
```
SELECT *
FROM patient
WHERE r_id BETWEEN 2 AND 5
```



```
    -> FROM patient
    -> WHERE r_id BETWEEN 2 AND 5;
+------+------+------+-----------+------------+----------+------------+
| p_id | r_id | d_id | p_name    | city       | contact  | p_date     |
+------+------+------+-----------+------------+----------+------------+
|  102 |    2 |  202 | Patient B | Tangail    | 01521001 | 2012-01-01 |
|  103 |    3 |  201 | Patient C | Rajshahi   | 0152147  | 2013-01-01 |
|  104 |    4 |  203 | Patient D | Mymensingh | 01520111 | 2014-01-01 |
|  105 |    5 |  203 | Patient E | Chandpur   | 0152146  | 2017-01-01 |
+------+------+------+-----------+------------+----------+------------+
4 rows in set (0.01 sec)

mysql>
```

Display the patients details according to their joining dates.

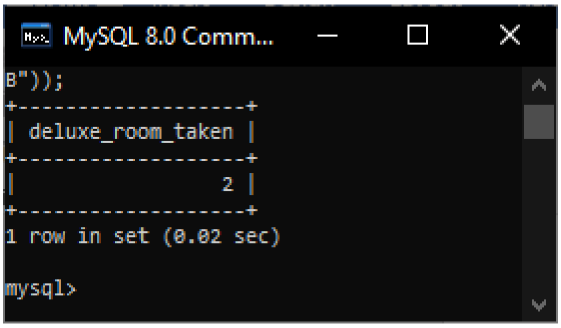
```
SELECT *

FROM patient

ORDER BY p_date DESC
```



```
    -> ORDER BY p_date DESC;
+------+------+------+-----------+------------+----------+------------+
| p_id | r_id | d_id | p_name    | city       | contact  | p_date     |
+------+------+------+-----------+------------+----------+------------+
|  105 |    5 |  203 | Patient E | Chandpur   | 0152146  | 2017-01-01 |
|  104 |    4 |  203 | Patient D | Mymensingh | 01520111 | 2014-01-01 |
|  103 |    3 |  201 | Patient C | Rajshahi   | 0152147  | 2013-01-01 |
|  102 |    2 |  202 | Patient B | Tangail    | 01521001 | 2012-01-01 |
|  108 |    8 |  201 | Patient H | Dilhi      | 01589    | 2010-09-01 |
|  110 |   10 |  206 | PatientJA | Mangalore  | 0821     | 2010-07-01 |
|  107 |    7 |  204 | Patient G | Kolkata    | 046421   | 2010-05-01 |
|  109 |    9 |  205 | Patient I | Karnataka  | 01571    | 2010-04-01 |
|  106 |    6 |  201 | Patient F | Cumilla    | 01146461 | 2010-02-01 |
|  101 |    1 |  201 | Patient A | Dhaka      | 01521    | 2010-01-01 |
+------+------+------+-----------+------------+----------+------------+
```

Count the patients who took deluxe rooms.

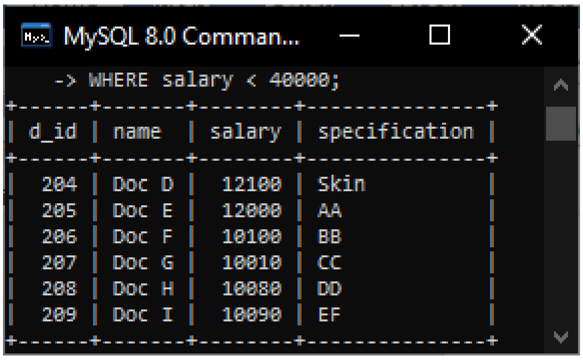
```
SELECT count(*) AS deluxe_room_taken

FROM patient

WHERE r_id IN (

    SELECT r_id

    FROM room

    WHERE room_type IN ("Room A", "Room B"))
```



```
B"));
+-------------------+
| deluxe_room_taken |
+-------------------+
|                 2 |
+-------------------+
1 row in set (0.02 sec)

mysql>
```

Display name of the doctor with salary less than 40000

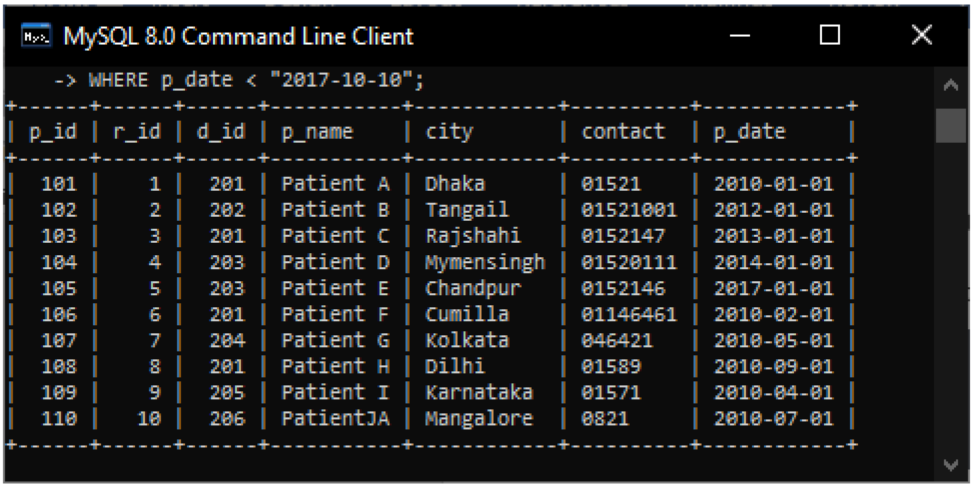
```
SELECT *

FROM doctors

WHERE salary < 40000
```



```
    -> WHERE salary < 40000;
+------+-------+--------+---------------+
| d_id | name  | salary | specification |
+------+-------+--------+---------------+
|  204 | Doc D |  12100 | Skin          |
|  205 | Doc E |  12000 | AA            |
|  206 | Doc F |  10100 | BB            |
|  207 | Doc G |  10010 | CC            |
|  208 | Doc H |  10080 | DD            |
|  209 | Doc I |  10090 | EF            |
+------+-------+--------+---------------+
```

Display the patients joined before 10.10.2017.

```
SELECT *

FROM patient

WHERE p_date < "2017-10-10"
```

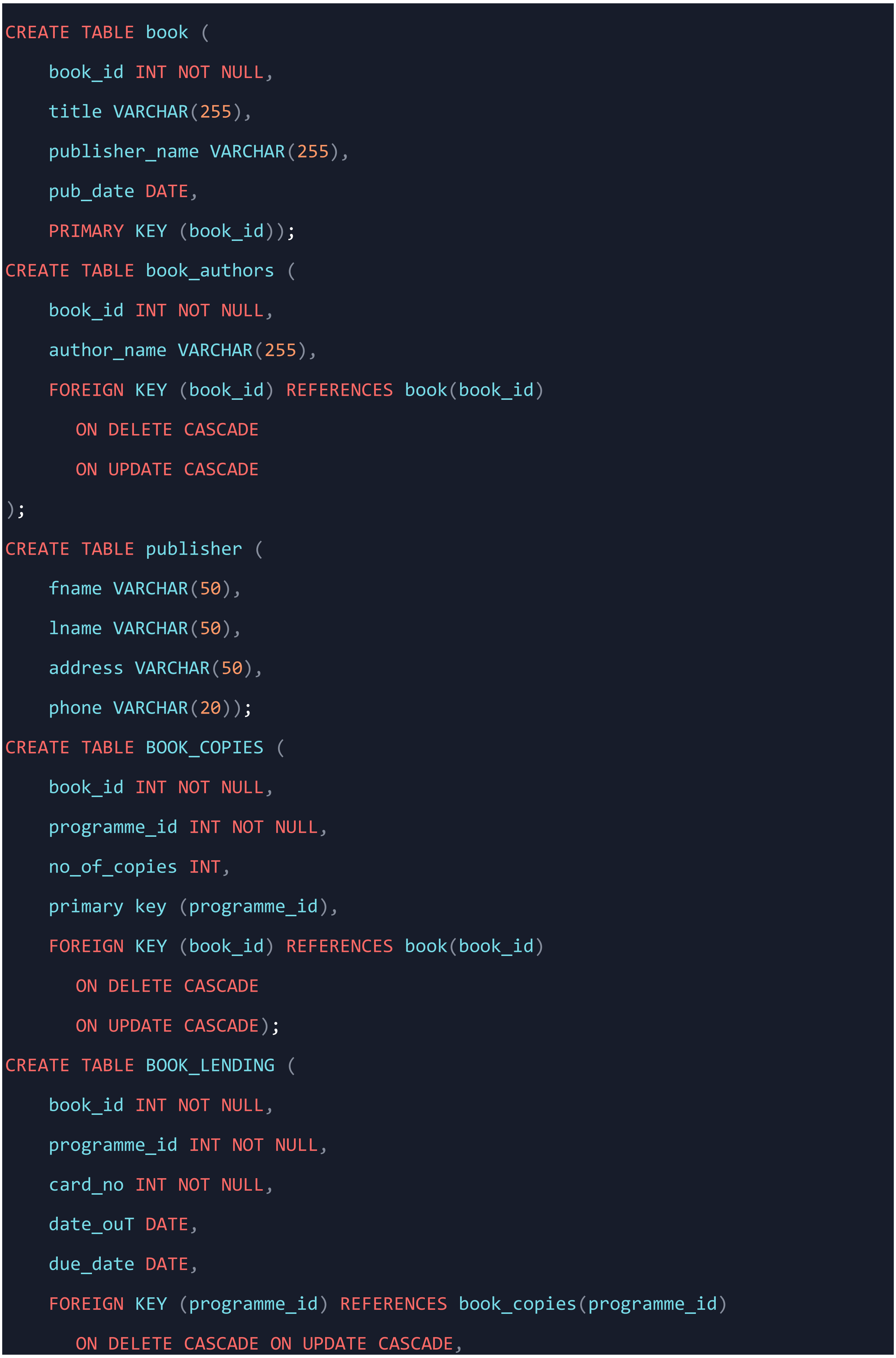


```
    -> WHERE p_date < "2017-10-10";
+------+------+------+-----------+------------+----------+------------+
| p_id | r_id | d_id | p_name    | city       | contact  | p_date     |
+------+------+------+-----------+------------+----------+------------+
| 101  |    1 |  201 | Patient A | Dhaka      | 01521    | 2010-01-01 |
| 102  |    2 |  202 | Patient B | Tangail    | 01521001 | 2012-01-01 |
| 103  |    3 |  201 | Patient C | Rajshahi   | 0152147  | 2013-01-01 |
| 104  |    4 |  203 | Patient D | Mymensingh | 01520111 | 2014-01-01 |
| 105  |    5 |  203 | Patient E | Chandpur   | 0152146  | 2017-01-01 |
| 106  |    6 |  201 | Patient F | Cumilla    | 01146461 | 2010-02-01 |
| 107  |    7 |  204 | Patient G | Kolkata    | 046421   | 2010-05-01 |
| 108  |    8 |  201 | Patient H | Dilhi      | 01589    | 2010-09-01 |
| 109  |    9 |  205 | Patient I | Karnataka  | 01571    | 2010-04-01 |
| 110  |   10 |  206 | PatientJA | Mangalore  | 0821     | 2010-07-01 |
+------+------+------+-----------+------------+----------+------------+
```

Create database for below Schema
(Add 10 entries for each table)

BOOK (Book_id, Title, Publiser_name, pub_date)
BOOK_AUTHORS (book_id, author_name)
PUBLISHER (fname, lname, address, phone)
BOOK_COPIES (book_id, programme_id, no_of_copies)
BOOK_LENDING (book_id, programme_id, card_no, date_out, due_date)

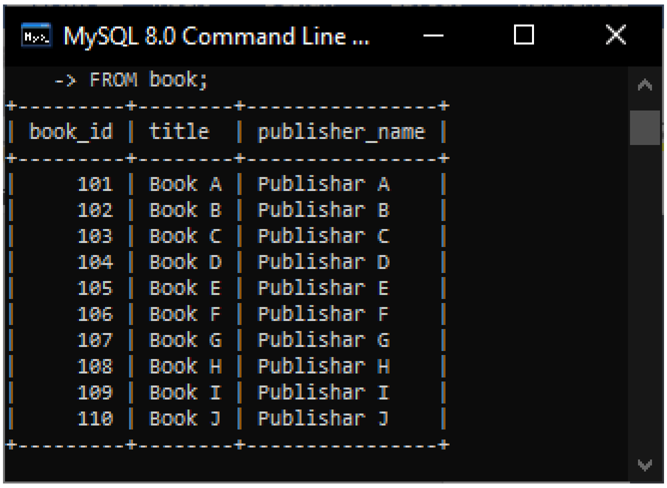
```
CREATE DATABASE library;

USE library;
```



```sql
CREATE TABLE book (
    book_id INT NOT NULL,
    title VARCHAR(255),
    publisher_name VARCHAR(255),
    pub_date DATE,
    PRIMARY KEY (book_id));
CREATE TABLE book_authors (
    book_id INT NOT NULL,
    author_name VARCHAR(255),
    FOREIGN KEY (book_id) REFERENCES book(book_id)
      ON DELETE CASCADE
      ON UPDATE CASCADE
);
CREATE TABLE publisher (
    fname VARCHAR(50),
    lname VARCHAR(50),
    address VARCHAR(50),
    phone VARCHAR(20));
CREATE TABLE BOOK_COPIES (
    book_id INT NOT NULL,
    programme_id INT NOT NULL,
    no_of_copies INT,
    primary key (programme_id),
    FOREIGN KEY (book_id) REFERENCES book(book_id)
      ON DELETE CASCADE
      ON UPDATE CASCADE);
CREATE TABLE BOOK_LENDING (
    book_id INT NOT NULL,
    programme_id INT NOT NULL,
    card_no INT NOT NULL,
    date_ouT DATE,
    due_date DATE,
    FOREIGN KEY (programme_id) REFERENCES book_copies(programme_id)
      ON DELETE CASCADE ON UPDATE CASCADE,
    FOREIGN KEY (book_id) REFERENCES book(book_id)
      ON DELETE CASCADE ON UPDATE CASCADE);

INSERT INTO book
    VALUES (101, "Book A", "Publishar A", "2017-01-01"),
    (102, "Book B", "Publishar B", "2013-01-01"),
    (103, "Book C", "Publishar C", "2014-01-01"),
    (104, "Book D", "Publishar D", "2015-01-01"),
    (105, "Book E", "Publishar E", "2012-01-01"),
    (106, "Book F", "Publishar F", "2011-01-01"),
    (107, "Book G", "Publishar G", "2010-01-01"),
    (108, "Book H", "Publishar H", "2003-01-01"),
    (109, "Book I", "Publishar I", "2022-01-01"),
    (110, "Book J", "Publishar J", "2021-01-01");
INSERT INTO book_authors
    VALUES (101, "Author A"),
    (102, "Author B"),
    (103, "Author C"),
    (104, "Author D"),
    (105, "Author E"),
    (106, "Author F"),
    (107, "Author G"),
    (108, "Author H"),
    (109, "Author I"),
    (110, "Author J");
INSERT INTO publisher
        VALUES ("Pub First A", "Pub Last A", "Address A", "012"),
        ("Pub First B", "Pub Last B", "Address B", "013"),
        ("Pub First C", "Pub Last C", "Address C", "014"),
        ("Pub First D", "Pub Last D", "Address D", "015"),
        ("Pub First E", "Pub Last E", "Address E", "016"),
        ("Pub First F", "Pub Last F", "Address F", "017"),
        ("Pub First G", "Pub Last G", "Address G", "018"),
        ("Pub First H", "Pub Last H", "Address H", "019"),
        ("Pub First I", "Pub Last I", "Address I", "010"),
        ("Pub First J", "Pub Last J", "Address J", "011");
INSERT INTO BOOK_COPIES
    VALUES (101, 201, 5), (102, 202, 4), (103, 203, 7), (104, 204, 8),
    (105, 205, 23), (106, 206, 1),
    (107, 207, 0),
    (108, 208, 10),
    (109, 209, 53),
    (110, 210, 50);
INSERT INTO book_lending
    VALUES (101, 201, 301, "2017-01-01", "2017-02-01"),
    (102, 202, 302, "2018-01-01", "2018-02-01"),
    (103, 203, 303, "2019-01-01", "2019-02-01"),
    (104, 204, 304, "2020-01-01", "2020-02-01"),
    (105, 205, 305, "2021-01-01", "2021-02-01"),
    (106, 206, 306, "2022-01-01", "2022-02-01"),
    (107, 207, 307, "2016-01-01", "2016-02-01"),
    (108, 208, 308, "2015-01-01", "2015-02-01"),
    (109, 209, 308, "2014-01-01", "2014-02-01"),
    (110, 210, 310, "2013-01-01", "2013-02-01");
```

Retrieve details of all books in the library – id, title, name of publisher.

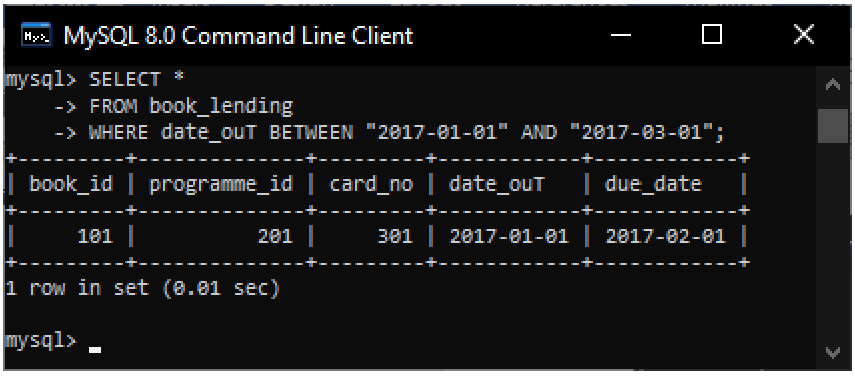
```
SELECT book_id, title, publisher_name
FROM book;
```



```
    -> FROM book;
+---------+--------+----------------+
| book_id | title  | publisher_name |
+---------+--------+----------------+
|     101 | Book A | Publishar A    |
|     102 | Book B | Publishar B    |
|     103 | Book C | Publishar C    |
|     104 | Book D | Publishar D    |
|     105 | Book E | Publishar E    |
|     106 | Book F | Publishar F    |
|     107 | Book G | Publishar G    |
|     108 | Book H | Publishar H    |
|     109 | Book I | Publishar I    |
|     110 | Book J | Publishar J    |
+---------+--------+----------------+
```

Retrieve the books which have been borrowed from Jan 2017 to March 2017

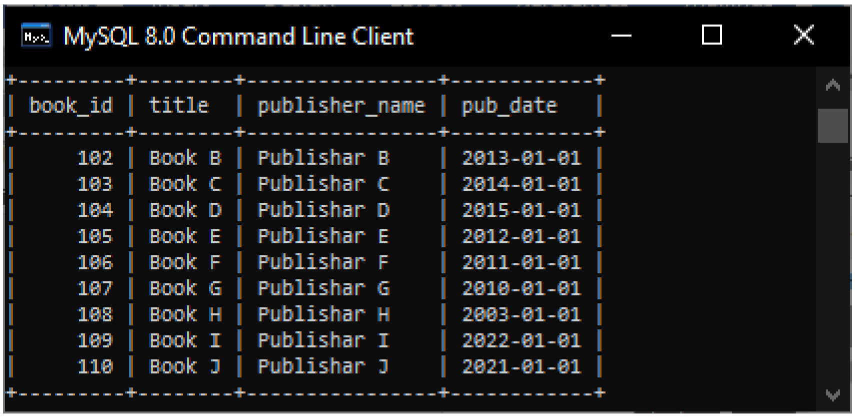
```
SELECT *

FROM book_lending

WHERE date_ouT BETWEEN "2017-01-01" AND "2017-03-01";
```



```
mysql> SELECT *
    -> FROM book_lending
    -> WHERE date_ouT BETWEEN "2017-01-01" AND "2017-03-01";
+---------+--------------+---------+------------+------------+
| book_id | programme_id | card_no | date_ouT   | due_date   |
+---------+--------------+---------+------------+------------+
|     101 |          201 |     301 | 2017-01-01 | 2017-02-01 |
+---------+--------------+---------+------------+------------+
1 row in set (0.01 sec)

mysql> _
```

Delete a book in the BOOK table. Update the contents of other tables to reflect this data manipulation operation.

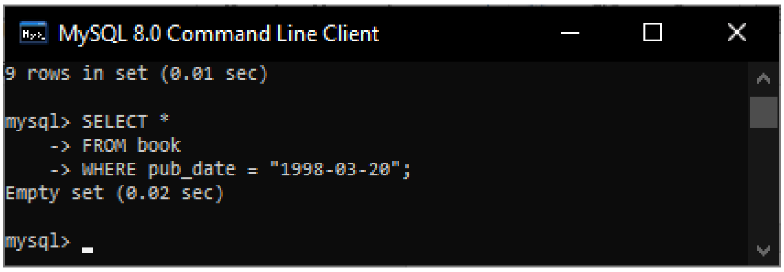
```
DELETE FROM book WHERE book_id = 101
```



```
+---------+--------+----------------+------------+
| book_id | title  | publisher_name | pub_date   |
+---------+--------+----------------+------------+
|     102 | Book B | Publishar B    | 2013-01-01 |
|     103 | Book C | Publishar C    | 2014-01-01 |
|     104 | Book D | Publishar D    | 2015-01-01 |
|     105 | Book E | Publishar E    | 2012-01-01 |
|     106 | Book F | Publishar F    | 2011-01-01 |
|     107 | Book G | Publishar G    | 2010-01-01 |
|     108 | Book H | Publishar H    | 2003-01-01 |
|     109 | Book I | Publishar I    | 2022-01-01 |
|     110 | Book J | Publishar J    | 2021-01-01 |
+---------+--------+----------------+------------+
```

Retrieve the details of the books(id, title, publisher name, year) published on the date 20-03-1998

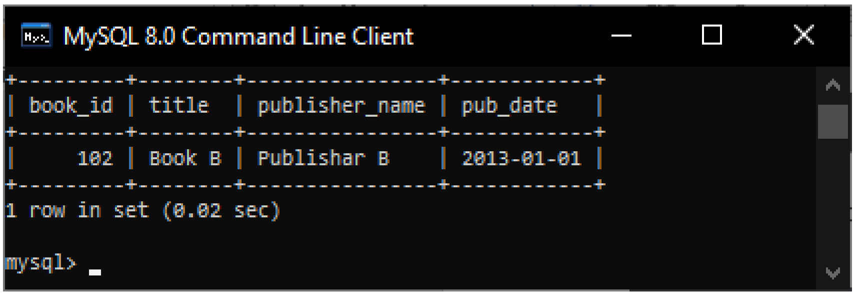
```
SELECT *

FROM book

WHERE pub_date = "1998-03-20"
```



```
9 rows in set (0.01 sec)

mysql> SELECT *
    -> FROM book
    -> WHERE pub_date = "1998-03-20";
Empty set (0.02 sec)

mysql> _
```

Retrieve the books published by a particular author.

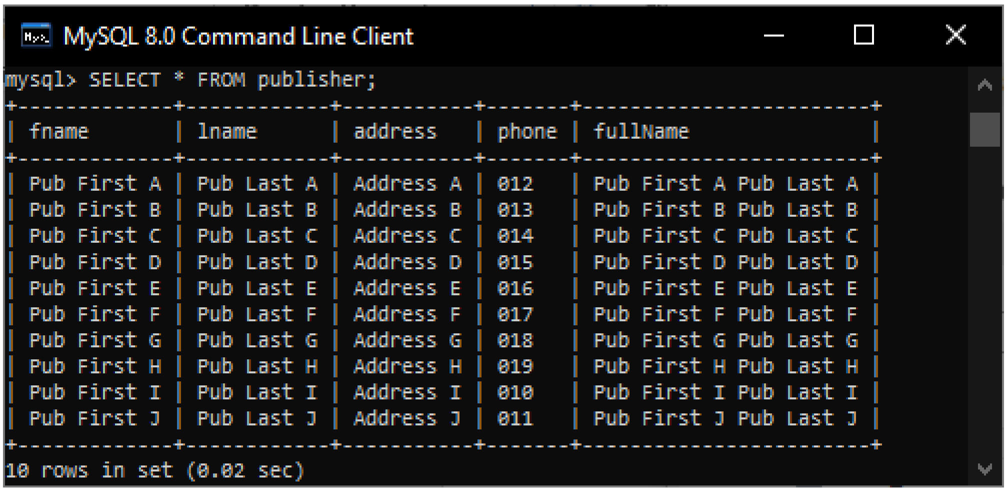
```
SELECT *
FROM book
WHERE book_id = (
    SELECT book_id
    FROM book_authors
    WHERE author_name = "Author B")
```



```
+---------+--------+----------------+------------+
| book_id | title  | publisher_name | pub_date   |
+---------+--------+----------------+------------+
|     102 | Book B | Publishar B    | 2013-01-01 |
+---------+--------+----------------+------------+
1 row in set (0.02 sec)

mysql>
```

Create a column name in the Publishers table. Combine FName and LName and print it in column name.

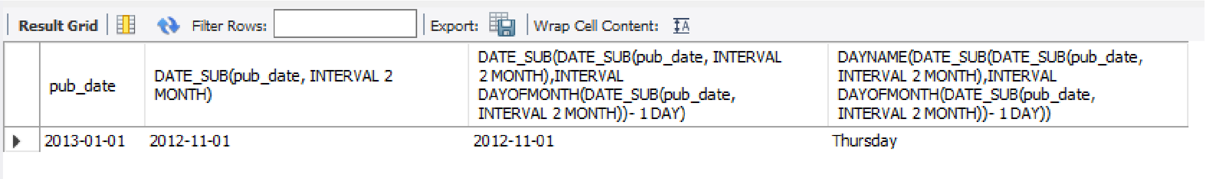
```
ALTER TABLE publisher
ADD fullName VARCHAR(50);
UPDATE publisher
SET fullName = concat(fname," ",lname);
```



```
mysql> SELECT * FROM publisher;
+-------------+------------+-----------+-------+-----------------------+
| fname       | lname      | address   | phone | fullName              |
+-------------+------------+-----------+-------+-----------------------+
| Pub First A | Pub Last A | Address A | 012   | Pub First A Pub Last A |
| Pub First B | Pub Last B | Address B | 013   | Pub First B Pub Last B |
| Pub First C | Pub Last C | Address C | 014   | Pub First C Pub Last C |
| Pub First D | Pub Last D | Address D | 015   | Pub First D Pub Last D |
| Pub First E | Pub Last E | Address E | 016   | Pub First E Pub Last E |
| Pub First F | Pub Last F | Address F | 017   | Pub First F Pub Last F |
| Pub First G | Pub Last G | Address G | 018   | Pub First G Pub Last G |
| Pub First H | Pub Last H | Address H | 019   | Pub First H Pub Last H |
| Pub First I | Pub Last I | Address I | 010   | Pub First I Pub Last I |
| Pub First J | Pub Last J | Address J | 011   | Pub First J Pub Last J |
+-------------+------------+-----------+-------+-----------------------+
10 rows in set (0.02 sec)
```

Write a query to display the first day of the month (in datetime format) two months before the current month from the date of publication of the book “DBMS”

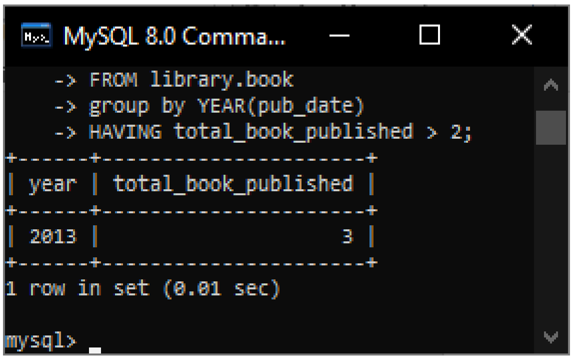
```
SELECT pub_date,
    DATE_SUB(pub_date, INTERVAL 2 MONTH),
    DATE_SUB(DATE_SUB(pub_date, INTERVAL 2 MONTH),INTERVAL
DAYOFMONTH(DATE_SUB(pub_date, INTERVAL 2 MONTH))- 1 DAY),
    DAYNAME(DATE_SUB(DATE_SUB(pub_date, INTERVAL 2 MONTH),INTERVAL
DAYOFMONTH(DATE_SUB(pub_date, INTERVAL 2 MONTH))- 1 DAY))

FROM book
WHERE title = "Book B"
```



pub_date	DATE_SUB(pub_date, INTERVAL 2 MONTH)	DATE_SUB(DATE_SUB(pub_date, INTERVAL 2 MONTH),INTERVAL DAYOFMONTH(DATE_SUB(pub_date, INTERVAL 2 MONTH))- 1 DAY)	DAYNAME(DATE_SUB(DATE_SUB(pub_date, INTERVAL 2 MONTH),INTERVAL DAYOFMONTH(DATE_SUB(pub_date, INTERVAL 2 MONTH))- 1 DAY))
2013-01-01	2012-11-01	2012-11-01	Thursday

Write a query to get years in which more than 3 books were published.

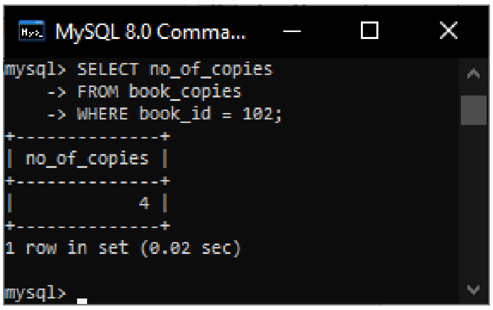
```
SELECT YEAR(pub_date) as year, count(*) as total_book_published
FROM library.book
group by YEAR(pub_date)
HAVING total_book_published > 2;
```



```
    -> FROM library.book
    -> group by YEAR(pub_date)
    -> HAVING total_book_published > 2;
+------+----------------------+
| year | total_book_published |
+------+----------------------+
| 2013 |                    3 |
+------+----------------------+
1 row in set (0.01 sec)

mysql> _
```

Print the number of copies of a particular book.

```
SELECT no_of_copies FROM book_copies WHERE book_id = 102;
```



```
mysql> SELECT no_of_copies
    -> FROM book_copies
    -> WHERE book_id = 102;
+--------------+
| no_of_copies |
+--------------+
|            4 |
+--------------+
1 row in set (0.02 sec)

mysql> _
```